DEEP LEARNING BASED FLOATING DEBRIS DETECTION IN MARINE ENVIRONMENT

A Project Report

Submitted to the APJ Abdul Kalam Technological University

in partial fulfillment of requirements for the award of degree

Bachelor of Technology

in

Electronics and Communication Engineering

by

POOJA JOSEPH(NSS20EC066)

MINNA MARIA(NSS20EC059)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NSS COLLEGE OF ENGINEERING, PALAKKAD

KERALA

MAY 2024

DEPT. OF ELECTRONICS & COMMUNICATION ENGINEERING

NSS COLLEGE OF ENGINEERING, PALAKKAD

2023-24

CERTIFICATE

This is to certify that the report entitled **DEEP LEARNING BASED FLOATING DEBRIS DETECTION IN MARINE ENVIRONMENT** submitted by **POOJA JOSEPH** (NSS20EC066) and **MINNA MARIA** (NSS20EC059) to the APJ Abdul Kalam Technological University in partial fulfillment of the B.Tech. degree in Electronics and Communication Engineering is a bonafide record of the project work carried out by them under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Dr. M Sabarimalai Manikandan	**Dr. Anil Kumar K R**	**Vinod G**
(External Guide)	(Internal Guide cum Coordinator)	(HOD)
Associate Professor	Associate Professor	Associate Professor
ECE Department	ECE Department	ECE Department
IIT Palakkad	NSSCE Palakkad	NSSCE Palakkad

DECLARATION

We hereby declare that the project report **DEEP LEARNING BASED FLOATING DEBRIS DETECTION IN MARINE ENVIRONMENT**, submitted for partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by us under supervision of **Dr. M Sabarimalai Manikandan**, Associate Professor, IIT Palakkad and **Dr. Anil Kumar K R**, Associate Professor. This submission represents our ideas in our own words and where ideas or words of others have been included, we have adequately and accurately cited and referenced the original sources.

We also declare that we have adhered to the ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Palakkad

01 MAY 2024

POOJA JOSEPH

MINNA MARIA

Abstract

This is an experimental study approach for detecting floating debris in water bodies using deep learning. Traditional methods for debris detection often rely on manual inspection or expensive remote sensing technologies, which are time-consuming and resource-intensive. In contrast, the proposed system uses deep learning to automate debris detection tasks.

The proposed framework consists of three main stages: data collection, model training, and inference. Initially, a diverse dataset of water surface images containing various types and sizes of debris is collected from online depositories or other suitable platforms. Subsequently, the YOLOv3 model is trained on this dataset to learn discriminative features of floating debris. To enhance the model's performance, various techniques like random rotations, flips, and scaling were employed to increase the diversity of training samples.

During the inference stage, the trained YOLOv3 model is utilized for real-time detection of floating debris in input images. The model efficiently localizes and classifies debris instances with high accuracy. Additionally, the proposed system can be integrated with autonomous vehicles or robotic platforms for autonomous debris collection, further enhancing its utility in practical scenarios.

Acknowledgement

We take this opportunity to express our deepest sense of gratitude and sincere thanks to everyone who helped us to complete this work successfully. We express our sincere gratitude to Dr. P.R Suresh, Principal, NSS College of Engineering, Palakkad for providing all the necessary facilities and support.

We express our sincere thanks to Dr. M Sabarimalai Manikandan, Associate Professor , IIT Palakkad for the guidance and support.

We would like to express our sincere gratitude to our project guide Dr. Anil Kumar K R, Associate Professor, Department of Electronics and Communication Engineering, NSS College of Engineering Palakkad for the guidance and mentorship throughout this work.

We would like to place on record our sincere gratitude to our project co-ordinator Dr. Sidharth N ,Associate Professor, Electronics and Communication Engineering,
NSS College of Engineering for his continuous support and coordination, which played a vital role in the completion of the project.

We would also like to place on record our sincere gratitude to Vinod G, Head of the Department, Electronics and Communication Engineering, NSS College of Engineering for providing all the necessary facilities and support.

Finally we thank our family, and friends who contributed to the succesful fulfilment of this project work.

POOJA JOSEPH
MINNA MARIA

Contents

Abbreviations

YOLO: You Only Look Once

YOLOv3: You Only Look Once version 3

UAV: Unmanned Aerial Vehicle

UUV: Unmanned underwater Vehicle

GUI: Graphical user interface

BRDF: Bidirectional Reflectance Distribution Function

SAD: Spectral Angle Distance

ROS: Robot Operating System

TSM: Total Suspended Matter

GPGP: Great Pacific Garbage Patch

PFAA: Perfluoroalkyl acid

FPN: Feature Pyramid Network

CNN: Convolutional Neural Network

NMS: Non-Maximal Suppression

IoU: Intersection over union

CBAM: Convolutional Block Attention Module

List of Figures

Chapter 1

Introduction

Marine debris, comprising solid materials introduced into aquatic environments intentionally or unintentionally, presents a critical environmental challenge. The increasing presence of floating debris in marine environments poses a significant threat to aquatic ecosystems and maritime activities worldwide. Plastic is the primary contributor to marine debris due to its slow degradation and lightweight properties, leading to the formation of expansive patches like the Great Pacific Garbage Patch (GPGP). Beyond the visual pollution, marine debris also poses severe threats to marine ecosystems and human health, as highlighted by the accumulation of perfluoroalkyl acid (PFAA) in seabirds near GPGP and its potential impact on the food chain. Traditional methods of debris detection often rely on manual inspection or expensive remote sensing technologies, which are time-consuming, costly, and lack real-time capabilities. These methods suffer from limitations in scope and efficiency, prompting the exploration of machine learning, particularly image segmentation, for automated detection. While existing studies have mainly focused on aerial photography, the utilization of satellite imagery remains underexplored, presenting a viable avenue for improving detection capabilities .

This project provides an experimental approach for automated floating debris detection using the You Only Look Once version 3 (YOLOv3) framework. Leveraging deep learning techniques, the proposed system offers realtime detection and classification of floating debris, enabling timely intervention and mitigation strategies. The model's performance has been evaluated on a diverse dataset of maritime images, demonstrating its robustness and effectiveness in detecting various types of debris under different environmental conditions, presenting an advanced level of detection with high accuracy and efficiency compared to current methods, demonstrating the potential of deep learning-based approaches for enhancing maritime environmental monitoring efforts.

1.1 Problem definition

The growing amount of floating debris in marine environments is a major concern for both aquatic ecosystems and maritime activities globally. Plastic, with its slow degradation and lightweight nature, is the primary contributor to marine debris. Traditional debris detection methods usually involve manual inspection or expensive remote sensing technologies, which are time-consuming, expensive, and lack real-time capabilities. These methods also have limitations in terms of scope and efficiency, which is why there's a growing interest in exploring machine learning for automated detection. Consequently, there's a pressing need for an efficient debris detection model that utilizes deep learning techniques to precisely detect and locate debris.

1.2 Objective

The main aime of the project is for developing a deep learning model using YOLOv3 for detection of debris in images. The specific objectives are to:

- Collect and preprocess the dataset
- Train the YOLOv3 model
- Evaluate the model

  The most important step is to gather a diverse dataset of images containing floating debris and annotate them with bounding boxes. Followed by Fine-tuneing the pre-trained YOLOv3 model on the annotated dataset to accurately detect floating debris and then assess its performance using a validation data and calculate relevant metrics.

1.3 Organization of the report

Chapter 1 describes the time-consuming nature and expense of conventional detection methods ultimately limiting their efficiency and scope. Consequently, it describes the main aim of the project to utilize YOLOv3, a deep learning technique, to detect floating debris in images, accompanied by specific objectives.

Chapter 2 summarizes and critically analyzes existing research, publications, and scholarly works relevant to the project. This aids in grasping the historical context, and fundamental principles pertinent involving different deep learning techniques. It helped in the identification of YOLOv3 as the suitable approach and its methodology. This chapter also deals with the existing methods for detecting and managing floating debris involving manual observation, remote sensing, and traditional cleanup methods, which are both labor-intensive and costly. This project aims to address these challenges by developing an automated debris detection model using deep learning technology, YOLOv3.

Chapter 4 introduces the the proposed system for detecting floating trash integrates YOLOv3, an advanced object detection algorithm, to overcome current limitations.It also provides information on YOLO and how it stands out from conventional methods. It involves the comparison between different versions of the network and the motivation to choose YOLOv3 and its architecture which predicts bounding boxes and class probabilities ensuring rapid processing.

Chapter 5 involves the different steps for implementation of the proposed system. It includes dataset collection , dataset annotation, preprocessing steps, training the model and testing it.

Chapter 6 gives idea about the results of the implemented model involving the output of the prediction algorithm and performance evaluation of the model using the loss curves and precision/recall graphs obtained.

Chapter 7 concludes the project as more precise than existing system to detect floating debris. Further optimization and fine-tuning of this model hold the potential to substantially contribute to environmental conservation, and mitigation of marine pollution.

Chapter 2

Literature review

This study [1] proposed a semi-automatic collection system for floating marine debris using a UAV and a UUV. Implemented with a robot operating system (ROS), it featured a graphical user interface (GUI) for target selection, UUV navigation control, and a collecting motion generator. Testing involved dynamic simulation and real robot trials. Results showed that UAV-UUV cooperation was more effective in debris collection compared to the UUV alone. The GUI demonstrated high usability, reducing workload, with an 88.23% success rate in usability. It had limitations like difficulty in converging to targets with small deviations and lack of testing in real marine environments.

The experimental approach [2] aimed to enhance marine monitoring using Sentinel optical data. They employed an unmixing-based data fusion technique, combining Sentinel 2 (S2) and Sentinel 3 (S3) images. S2 underwent bidirectional reflectance distribution function (BRDF) correction, while S3 provided hyperspectral data. Evaluations using correlation coefficient and spectral angle distance (SAD) showed improved resolution over S3, facilitating detailed estimations of chlorophyll-a (Chl-a) and total suspended matter (TSM) concentrations. Limitations included the dynamic marine environment and time lag between S2 and S3 acquisitions.

The intention of this paper [3] was to present a learning-based approach using satellite imagery to detect floating plastic debris in river. The used a compilation of a dataset of labeled images from the Sentinel-2 mission, adaptation of well-known image segmentation architectures (U-Net and DeeplabV3+), and the training and testing of these architectures using the dataset. This method showed 80% to 90% accuracy for debris pixels, and resulted in an IoU score of 0.3, 0.0023, and0.378 for the water, debris, and other pixels respectively.

The paper [4] introduced the FloW dataset, the first dataset for floating waste detection in inland waters, which consists of two sub-datasets: FloW-Img and FloW-RI.Methods used for implementation include deep learning-based object detection algorithms, image processing techniques for radar point cloud projection, and manual annotation of radar point clouds. It can be observed that there is a need for improvement in the robustness of vision-based object detection models for practical applications in floating waste detection.

The study [5] proposed TC-YOLOv5 object detection algorithm for recognizing floating debris. It involves integrating the Convolutional Block Attention Module (CBAM) and vision transformer in the YOLOv5n architecture to improve accuracy and reduce missed detections. The dataset includes seven categories of floating debris: bottle, branch, milk box, plastic bag, plastic garbage, ball, and leaf. It performed better than all other YOLOv4 and YOLOv5 series models in terms of accuracy, speed, resource utilization, and model size, making it suitable for practical applications. But it has model complexity and high cost of implementation.

A journal published in 2019 [6] aimed to develop a machine learning method for detecting and classifying floating macro-plastic debris in the ocean. through Convolutional Neural Network (CNN) approach. They trained the CNN on a dataset of images of plastic objects larger than a few centimeters, also known as macro-plastics. The trained model was then used to predict the class of new images of macro-plastic objects floating at sea.

The paper [7] implemented a hierarchical Bayesian modeling approach to estimate the detection probability and presence probability of mega-debris in the Mediterranean Sea. They used a strip-transect protocol and aerial surveys conducted in 2012 (SAMM survey) and 2018 (ASI survey) to collect data using a high-wing double-engine aircraft. The data recording was done using specific software during the surveys. The study validated their models using the SAMM survey data . They used Bayesian model evaluation techniques and assessed parameter convergence using statistical methods. The performance metrics used were means, standard deviations, and 80 % credibility intervals.

The experimental approach [8] proposed a system that leverages wireless signals to sense floating debris, with a focus on plastic debris in water bodies. The system uses microwave/millimeter wave sensors and directional antennas working in the 5 GHz band to detect variations in the wireless signals caused by floating debris. The received signal samples are averaged over a window of 2000 samples and plotted over 500 consecutive windows. Accuracy of the system was 95,5% and Specificity was 94.3%. Limitations of the proposed system include the need for a clear line of sight between the transmitting and receiving antennas, the potential for interference from other wireless signals, and the limited range of the system.

The primary objective of the paper [9] was to provide information on various technologies and approaches for the detection and monitoring of contaminants of emerging concern (CECs) and plastic debris in marine environments. The system used biosensors, specifically immunosensors, combined with a lab-on-a-chip configuration and amperometric detection for real-time analysis. The system was tested for the detection of three pesticides in seawater and compared to results from LC-MS analysis. The system also included an on-line solid-phase extraction module for improved sensitivity and robust measurements in real environmental conditions.

The paper[10] aims to develop an approach for automatically detecting and identifying floating marine debris using satellite imagery. The approach utilized multispectral satellite imagery from the Sentinel-2 mission. The satellites have a MultiSpectral Instrument (MSI) that captures data in 13 spectral bands ranging from visible to short-wave infrared. The proposed approach achieved high accuracy rates for classifying suspect plastic debris, with accuracy values above 90% for most classes. The limitations of the approach include the need for ground-truth validation, the limited applicability to other datasets and sensors, and the challenges posed by factors such as clouds, shallow waters, suspended sediment, and rough sea conditions.

The potential of using open remote sensing data and machine learning models for detecting and classifying marine floating plastics (MFP) in five different locations was explored in this paper[11]. Methods used includes Supervised machine learning models, specifically Support Vector Machine (SVM) and Random Forest (RF), for classification and prediction analysis. Sentinel-2 satellite imagery Remote sensing bands and spectral indices were used as features for developing the models. It had an accuracy of 91% with RF outperforming SVM in most cases with an overall accuracy of 88% for Beirut and 94% for Calabria.

Table 2.1: Literature Review

Author	**Year**	**Summary**
N. Shirakura et al. [1]	2021	Proposed a semi-automatic debris collection system using a UAV and a UUV, orchestrated through ROS, featuring a GUI for target selection, UUV navigation control, and collecting motion generation. Testing involved dynamic simulation and real robot trials, showing more effective debris collection compared to the UUV alone.
M. Kremezi and V. Karathanassi [2]	2020	Proposed an unmixing-based data fusion technique, combining Sentinel 2 (S2) and Sentinel 3 (S3) images to enhance marine monitoring. S2 underwent BRDF correction, and S3 provided hyperspectral data. Evaluation showed improved resolution over S3, enabling detailed estimations of Chl-a and TSM concentrations.
Àlex Solé Gómez et al. [3]	2022	Presented a learning approach using satellite imagery to detect floating plastic debris in rivers. They compiled a dataset of labeled images from Sentinel-2, adapted well-known image segmentation architectures (U-Net and DeeplabV3+), and trained and tested these architectures using the dataset. The method achieved 80-90% accuracy for debris pixels, with IoU scores of 0.3 for water, 0.0023 for debris, and 0.378 for other pixels.

Continued on next page

Table 2.1 – continued from previous page

Author	Year	Summary
Li et al. [4]	2023	Introduced the FloW dataset, consisting of two sub-datasets: FloW-Img and FloW-RI. Methods included deep learning-based object detection algorithms, image processing techniques for radar point cloud projection, and manual annotation of radar point clouds.
Y. Cheng et al. [5]	2021	Proposed the TC-YOLOv5 object detection algorithm for recognizing floating debris. They integrated the Convolutional Block Attention Module (CBAM) and vision transformer in the YOLOv5n architecture to improve accuracy and reduce missed detections. The dataset included seven categories of floating debris. TC-YOLOv5 outperformed all other YOLOv4 and YOLOv5 models,but it have high implementation cost.
Kylili, K., Kyriakides, I., Artusi, A. et al. [6]	2019	Devised a machine learning method using a Convolutional Neural Network (CNN) to detect and classify macro-plastic debris in the ocean. Trained on a dataset of images, the CNN achieved a validation accuracy of approximately 86%.
C. Lambert, M. Authier, G. Dorémus et al. [7]	2020	Employed hierarchical Bayesian modeling to estimate detection and presence probabilities of mega-debris in the Mediterranean Sea. Utilized strip-transect protocols and aerial surveys in 2012 and 2018 to validate models, considering factors like sea state and glare severity.

Continued on next page

Table 2.1 – continued from previous page

Author	**Year**	**Summary**
A. Mahanti, A. Banerjee and K. Srinivasan [8]	2021	Developed a system to detect floating debris, particularly plastic debris in water, using wireless signals. Employed microwave/millimeter-wave sensors and directional antennas in the 5 GHz band, detecting signal variations caused by debris. Achieved 95.5% accuracy and 94.3% specificity, implemented on a WARP v3 node with a Xilinx Zyng-7020 FPGA and ARM Cortex-A9 processor.
Marinella and Farré [9]	2020	Explored technologies for detecting contaminants of emerging concern (CECs) and plastic debris in marine environments. Highlighted the Sea-on-a-Chip project, which developed an on-site analytical system for real-time monitoring of CECs using immunosensors and amperometric detection.
M. M. Duarte and L. Azevedo et al. [10]	2023	Developed an approach using multispectral satellite imagery from the Sentinel-2 mission. Employed extreme gradient boosting as the classification algorithm, achieving accuracy rates exceeding 90% for suspect plastic debris classes.
Srikanta Sannigrahi, Bidroha Basu et al. [11]	2022	Delved into utilizing open remote sensing data and machine learning models, specifically Support Vector Machine (SVM) and Random Forest (RF), for detecting and classifying marine floating plastics (MFP) across five locations. Employed hyperparameter tuning and atmospheric correction processors, achieving 91% accuracy, with RF surpassing SVM.

2.1 Existing systems

The present state of environmental monitoring and management involves detecting and addressing floating trash primarily through manual observation, remote sensing technologies, and traditional cleanup methods. Although these approaches have provided valuable insights and interventions, they are hindered by their dependence on human labor, expensive equipment, and reactive tactics.

2.1.1 Manual observation

One of the primary methods for detecting floating debris is through manual observation by field personnel or volunteers. However, this method is labor-intensive, time-consuming, and prone to errors. Human observers can only cover limited areas, and their ability to detect floating debris may vary based on factors such as weather conditions, visibility, and observer experience. As a result, many instances of floating trash may go undetected, leading to environmental degradation and potential hazards to marine life and human activities.

Advantages

- Provides valuable insights into localized instances of floating debris.
- Can be used in areas where technological solutions are not feasible.

Disadvantages

- Labor-intensive and time-consuming.
- Prone to errors and incomplete coverage.
- Limited coverage area and effectiveness dependent on observer experience and environmental conditions.

2.1.2 Remote sensing technologies

Satellite imagery and aerial surveys are used to monitor large-scale patterns of marine pollution. While these tools offer useful data on the extent and distribution of floating trash, they suffer from limited spatial and temporal resolution, making them less effective for localized monitoring and real-time detection.

Advantages

- Provides data on large-scale patterns of marine pollution.
- Useful for monitoring the extent and distribution of floating trash over large areas.

Disadvantages

- Limited spatial and temporal resolution.
- Less effective for localized monitoring and real-time detection.
- May not detect smaller debris or accurately pinpoint the location of debris in real-time, especially in areas with complex coastal or marine environments.

2.1.3 Traditional cleanup methods

Traditional cleanup operations involve deploying boats, nets, and other equipment to physically remove floating trash from water bodies. While these efforts alleviate immediate pollution impacts, they tend to be reactive and fail to address pollution's root causes or prevent future occurrences.

Advantages

- Alleviates immediate pollution impacts.
- Removes visible debris from the surface of the water.

Disadvantages

- Reactive and fails to address root causes of pollution.
- Costly, resource-intensive, and not sustainable in the long term.
- Neglects smaller or submerged debris that may pose significant environmental threats.

Overall, the current system for detecting and mitigating floating trash in aquatic environments is characterized by its reliance on manual labor, restricted technological capabilities, and reactive strategies. This underscores the urgent need for innovative and automated solutions that can enhance the efficiency, accuracy, and timeliness of floating debris detection and management.

To overcome these challenges, this project develops an automated floating debris detection system using deep learning technology. Specifically, the project will utilize the YOLOv3 object detection algorithm to detect floating debris in images acquired using different sources. By utilizing the capabilities of deep learning, the system overcomes the limitations of existing methods thereby providing a more efficient and scalable model for monitoring and managing floating trash in aquatic environments.

Chapter 3

Proposed model

The proposed model for detecting floating trash is using YOLOv3, an advanced object detection algorithm, to address limitations in current methods through the integration of artificial intelligence.

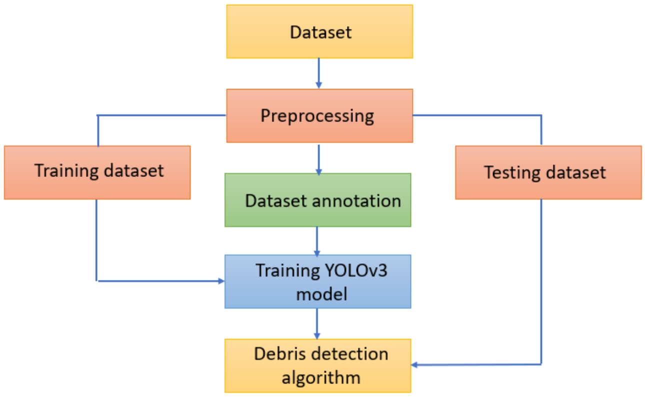

Figure 3.1: Proposed model-Flow diagram

Automated detection algorithm : The core component of the model is an automated detection algorithm powered by YOLOv3. This algorithm enables the automatic identification of floating trash in images, ensuring high accuracy and efficiency. By leveraging YOLOv3's real-time processing capabilities, the system can swiftly analyze large volumes of data, facilitating timely responses to potential pollution incidents. It includes the following steps.

Dataset creation and annotation : The development of this system requires the creation of a diverse, annotated dataset depicting floating trash in various aquatic environments.

This dataset is essential for training and validating the YOLOv3 model, ensuring its effectiveness across different types of debris. The process involves collecting images of water bodies polluted with floating debris and annotating them with bounding boxes around each piece of trash.

YOLOv3 model training : Once the annotated dataset is ready, the YOLOv3 model is trained using the deep learning techniques. During training, the model learns to recognize various types of floating trash. The training process involves optimizing parameters and hyperparameters to enhance detection accuracy and efficiency. By iteratively adjusting the model's parameters, the system ensures optimal performance in identifying floating debris.

Real-time detection : Once trained, the YOLOv3-based system is capable of analyzing images in real-time, annotating and it detects pollutants as trash.

3.1 YOLO

YOLO (You Only Look Once) is a groundbreaking object detection system, celebrated for its remarkable speed and precision. Diverging from conventional methods reliant on intricate pipelines, YOLO operates with a single pass through the network to detect objects in images. This streamlined approach not only ensures rapid performance but also positions YOLO as a leading choice among object detection algorithms, widely embraced for its efficiency and accuracy.Key Features of YOLO are:

1. Single Pass Detection : It processes images by a single pass through the neural network. Unlike traditional detection systems, it directly anticipates the class probabilities and bounding boxes from the entire image.

2. Grid-based Approach : It partitions the input image into a grid and predicts the bounding boxes directly from the entire image. This method based on a grid enables YOLO to detect multiple objects concurrently.

3. Real-time Performance : It process the images extremely fast, achieving real-time performance. It is suitable autonomous driving, robotics and surveillance,

Floating debris poses a significant threat to marine ecosystems, wildlife, and maritime activities. Detecting and monitoring floating debris in water bodies is essential for environmental protection, navigation safety, and maritime security. YOLO's detection capabilities make it an ideal solution for detection and tracking floating debris in water bodies.

3.2 Comparison of YOLO versions

Table 3.1: Comparison of YOLOv1, YOLOv2, and YOLOv3

Feature	**YOLOv1**	**YOLOv2**	**YOLOv3**
Backbone	Darknet-19	Darknet-19	Darknet-53
Pre-trained weights	No	Yes (ImageNet)	Yes (ImageNet)
Prediction	Single scale	Multi-scale	Multi-scale
Bounding Box	No anchor boxes	Anchor boxes	Anchor boxes
Architecture	24 conv + 2 FC layers	53 convolutional layers	53 convolutional layers
Objectness	Combined with classification loss	Predict objectness score directly	Predict objectness score directly
Training	End-to-end	End-to-end	Multi-scale training
Speed	Medium	Faster than YOLOv1	Faster than YOLOv2
Accuracy	Moderate	Better than YOLOv1	Better than YOLOv2

3.3 YOLOv3

YOLO is a state-of-the-art object detection algorithm that has popularity in various fields like computer vision. YOLOv3 (You Only Look Once version 3) represents the third evolution of the YOLO algorithm, offering improved speed, accuracy, and efficiency compared to its predecessors. Main features of YOLOv3 are given below:

1. Single Network Architecture: YOLOv3 embodies a unified neural network architecture that forecasts bounding boxes for numerous objects within an image. Unlike traditional object detection algorithms that use distinct networks for region

proposal and classification of objects, YOLOv3 performs both tasks in a single pass, making it more efficient.

2. Speed and Efficiency: YOLOv3 is known for its speed and efficiency, capable of processing images in real-time. By using a single pass to perform the detection, YOLOv3 eliminates the need for complex post-processing techniques, resulting in faster inference times.

3. Improved Accuracy: YOLOv3 improves upon the accuracy of its predecessors by incorporating various architectural changes and training techniques.It uses feature pyramid network (FPN) to extract various features, allowing it to detect objects of different sizes more effectively. YOLOv3 also employs multi-scale training, data augmentation, and other techniques to improve its accuracy in detecting small objects and objects with low contrast.

4. Flexibility and Versatility: YOLOv3 is a versatile algorithm that can be trained to detect a wide range of objects in different environments and under various conditions. It can detect objects of different shapes, sizes, and orientations, making it suitable for applications such as object detection, instance segmentation, and pose estimation.

5. Open Source and Easy to Use: YOLOv3 is open-source and freely available, making it accessible to researchers, developers, and practitioners. It is implemented in popular deep learning frameworks like Pytorch and TensorFlow, allowingto train and deploy custom object detection models easily.

6. Integration with Existing Systems: YOLOv3 can be seamlessly integrated with existing environmental monitoring and management systems, allowing for enhanced detection and management of floating debris in aquatic environments.

3.3.1 Architecture

In YOLOv3, a convolutional neural network (CNN) is utilized for detection of debris in images. The network is organized into three primary components: backbone, neck, and the head. The backbone consists of a series of convolutional layers aimed at extracting features from the input image. The neck then combines features from different scales to enhance the detection of objects. Finally, the head is comprised of fully connected layers that predict the class and location of each object in the

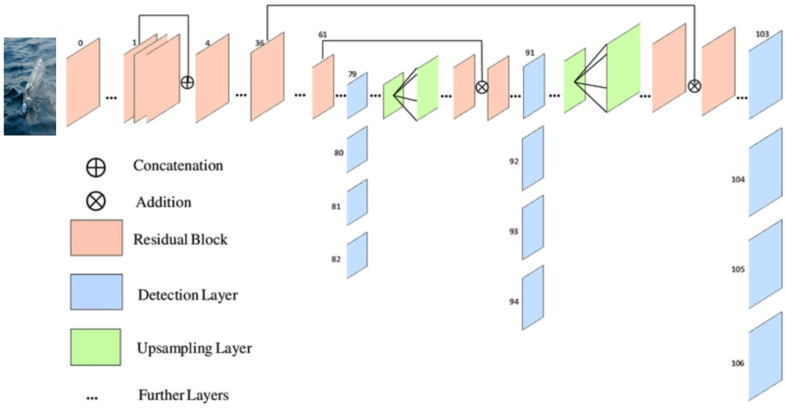

Figure 3.2: YOLOv3 network archiecture

image. Moreover, in YOLOv3, anchor boxes are integrated, representing pre-defined bounding boxes with diverse size and aspect ratio. These anchor boxes are crucial for accurately predicting the location and size of objects within the image. The YOLOv3 architecture is derived from Darknet-53, a model for feature extraction. It comprises 53 stacked layers, totaling 106 layers in a fully convolutional setup. Object detection is performed at three specific layers: the 82nd, 94th, and 106th layers. The stride is 32, 16, and 8 respectively. In the 53-layer architecture, every layer is followed by batch normalization layer and utilizes the Leaky ReLU activation function.

The core principle of the YOLOv3 architecture is partitioning images into cells sized SxS. Each cell is tasked with predicting objects within its confines. Hence, the 82nd layer generates a 13x13 output (416/32 = 13), dedicated to detecting large objects due to its 32 stride. Similarly, the 94th layer produces a 26x26 output (416/16 = 26) with a stride of 16, aimed at identifying medium-sized objects. Finally, the last layer (106th) has a stride of 8, allowing it to detect large objects, resulting in a 52x52 output (416/8 = 52).

Chapter 4

Model implementation

The proposed model has been implemented through the following steps:

4.1 Dataset collection

For the development of the system for debris detection, a dataset comprising approximately 2000 images were collected. These images were sourced from various online repositories and captured through physical photography in nearby water bodies.

The images were selected based on their relevance to the project's objectives, focusing on capturing a wide range of floating debris in diverse aquatic environments. Special attention was given to including images depicting different types of debris commonly found in marine ecosystems, such as plastics, bottles, bags, containers, and other waste materials.

Figure 4.1: Examples of dataset collected

A portion of the dataset was obtained from online repositories specializing in environmental photography and marine conservation. These repositories provided a valuable source of images depicting floating debris in various water bodies worldwide.

To supplement the online dataset, photographs were taken directly from nearby water bodies, including rivers, lakes, and coastal areas. This approach ensured the inclusion of images specific to the project's geographical location, enhancing the dataset's diversity and relevance.

4.2 Dataset annotation

The collected dataset was annotated to identify floating debris. Each image was examined, and bounding boxes were drawn around individual pieces of floating trash using LabelImg, which is a graphical image annotation tool which is in Python language and has Qt as its graphical interface.

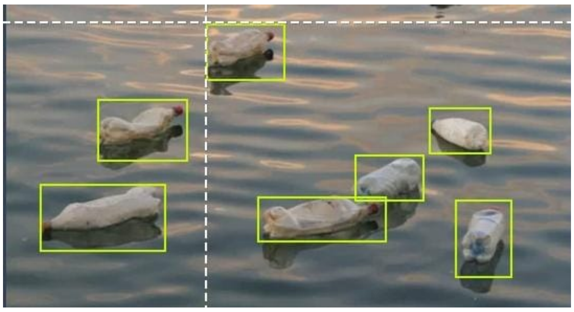

Figure 4.2: Image Annotation

The annotation process focused on identifying various types of floating debris commonly found in marine environments. This included plastic bottles, bags, containers, wrappers, and other waste materials. Strict quality control measures were implemented to ensure the accuracy and consistency of the annotations,thus providing labeled data for training and validating the deep learning model to accurately detect floating debris in marine environments.

4.3 Preprocessing

The collected dataset underwent preprocessing to standardize the image size. Each image was resized to a resolution of 416×416 pixels, ensuring uniformity across the dataset. YOLOv3 requires this size for its input images. The scale factor employed normalizes the pixel values of the input image. Dividing by 255 ensures that the

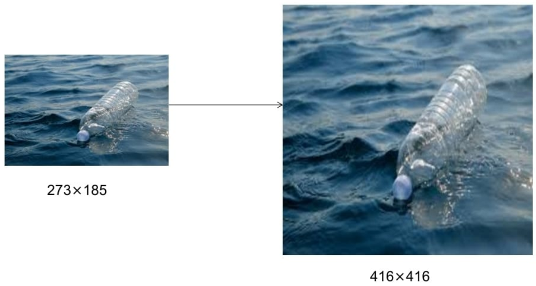

Figure 4.3: Resize image to resolution of 416×416

pixel values are within the range [0, 1], which is a common preprocessing step for YOLOv3. Furthermore, while OpenCV typically loads images in BGR (Blue, Green, Red) channel order, YOLO models are trained on images in RGB (Red, Green, Blue) channel order. Thus, adjustments are made to ensure that the input image is in the correct channel order for compatibility with the YOLO model. To enhance the robustness and generalization of the deep learning model, data augmentation was applied to the preprocessed data which include the following transformations:

Flipping: Images were horizontally flipped to create mirror images, effectively doubling the size of the dataset.

Rotating: Images were randomly rotated by various degrees to simulate different orientations of floating debris in marine environments. This helped in increasing the dataset diversity, making the model more robust to variations in debris orientation and appearance.

The preprocessing steps ensured that all images in the dataset were standardized to a consistent size and format, facilitating efficient training of the model. By resizing images to 416×416 pixels, the computational complexity of the model was reduced, while preserving important visual features of the debris.

4.4 Model training

Model training involves adjusting the parameters of the neural network of YOLOv3 to minimize the difference between its predictions and the ground truth labels in the annotated dataset.

The algorithm iteratively processes batches of annotated data, adjusting the weights and biases of the network based on the loss calculated between its predictions and the actual annotations. This process, often referred to as backpropagation, minimizes the variation between the actual and the predicted outputs, gradually improving performance of the model's performance.

The algorithm divides an image into N grids, with a uniform SxS dimensional region. Each grid is tasked with detecting and localizing objects within its allocated area. Within these grids, predictions are made for bounding box coordinates, alongside the labels and the probabilities of object presence.

Although this approach efficiently distributes detection and recognition tasks among image cells, multiple cells may predict the same object which has different bounding box predictions. To resolve this issue, non-maximal suppression (NMS) is employed. This will filter out bounding boxes with lower scores. Initially, it identifies the bounding box with the highest probability score. Then, it eliminates bounding boxes that significantly overlap with this highest-scored box, using intersection over union (IoU) calculations.

4.5 Model testing

The algorithm starts by importing essential libraries for numerical calculations, execution timing, image processing using OpenCV, and system interaction. It establishes constants like the confidence threshold and class labels, and generates random colors for visualizing bounding boxes.

Using OpenCV, the YOLOv3 model is loaded from the configuration file and weights file, providing the neural network model. It then proceeds to loop through each image file in the specified directory, passing each image through the YOLOv3 model. This step produces predictions for bounding box coordinates, class probabilities, and confidence scores.

Following this, the algorithm executes Non-Maximum Suppression (NMS) for eliminating redundant bounding boxes, so that every object is detected uniquely. Finally, it draws bounding boxes around the detected objects and displays the results.

Chapter 5

Result and discussion

The proposed model for floating debris detection using YOLOv3 has been implemented and the results has been obtained.

5.1 Model output

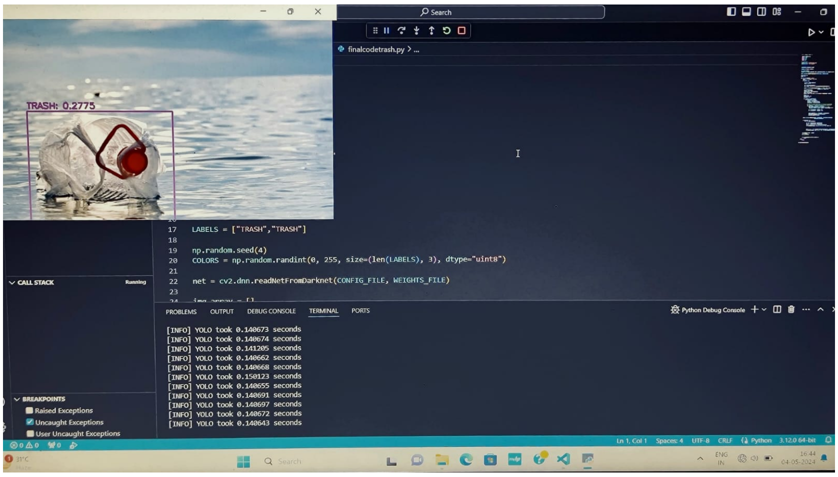

Figure 5.1: Debris detection output

YOLOv3 has successfully identified and located debris within the image. It draws bounding boxes around detected debris objects. These boxes indicate the exact location of each detected object in the image. Bounding box is associated with a class label of 'trash'. It also assigns a confidence score. This score reflects the algorithm's confidence level that the detected object is indeed debris. Higher scores indicate greater confidence in the detection.

5.2 Performance evaluation

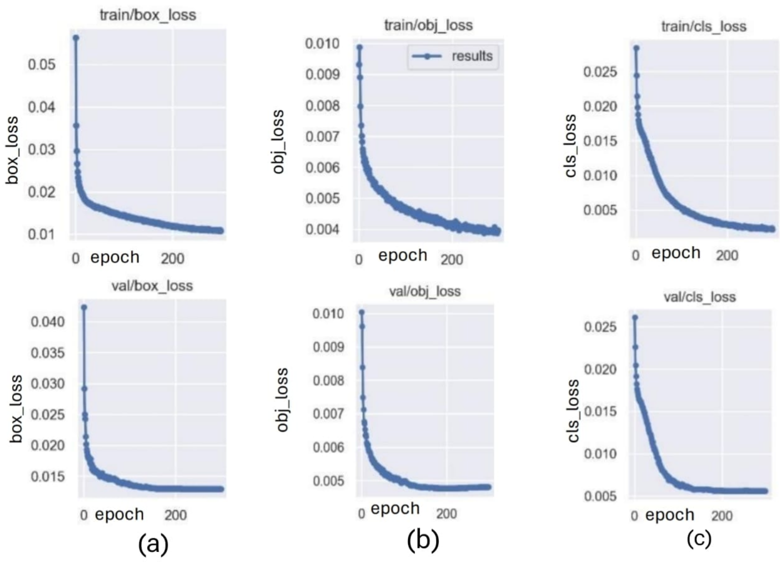

Figure 5.2: Graphs of (a) box loss, (b) object loss, (c) classification loss, of training and validation dataset respectively

Figure 8.1 shows the training and validation loss curves for the YOLOv3 object detection model. These curves illustrate the relationship between the number of epochs (x-axis) and the loss (y-axis). (a) Box loss likely represents the model's precision in predicting bounding boxes around objects in images. Lower box loss values suggest better accuracy in predicting object size and location. (b) Obj_loss is likely associated with the model's ability to distinguish between foreground objects and background clutter. Decreasing graph implies improved object discernment from the background. (c) Cls_loss presumably reflects the model's effectiveness in classifying detected debris as trash. Lower cls_loss values indicate improved accuracy in recognizing the correct object class.

Throughout the graphs, the solid lines represent the training loss, which starts at higher values and gradually decreases as the model undergoes more training epochs. This decline indicates the model's learning process and its enhancement in predictive capabilities. Conversely, the dashed lines represent the validation loss, which follows a similar trend but generally remains higher than the training loss. This discrepancy is because the validation loss is computed on a separate dataset that the model has not been trained on. Here the model generalizes well as the loss curves for both training and validation dataset show similar trends.

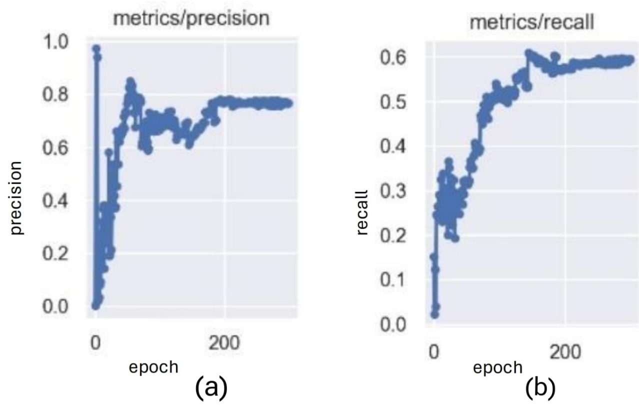

Figure 5.3: Graph of (a) Precision and (b) Recall

Figure 8.2 illustrates the precision and recall metrics for the model. Precision refers to the proportion of objects identified as debris by the model that were indeed debris and recall indicates the proportion of actual debris objects correctly identified by the detection model. High recall implies effective identification of most debris objects.

Precision graph (a) shows that the precision starts at 1.0, then gradually decreases as recall increases to 0.6. This suggests the model begins with high precision but misses many debris objects, leading to a decline in precision as recall improves. Similarly, in recall graph (b), recall begins at 0 and steadily increases to nearly 0.6 as precision decreases to 0.1, highlighting the trade-off between precision and recall. These graphs infer that the precision for the implemented model is high but the recall is low.

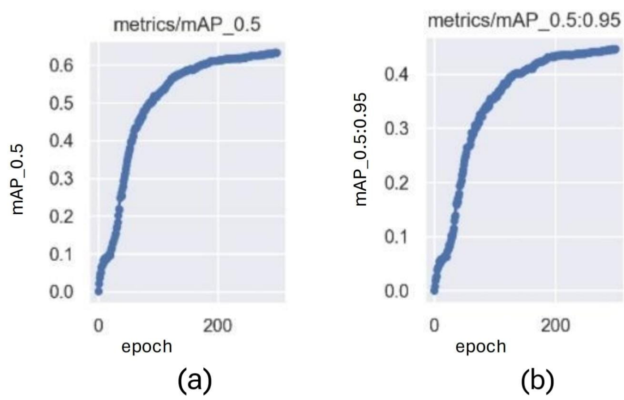

Figure 5.4: Graphs of (a) mAP_0.5 and (b) mAP_0.5:0.95

Figure 8.3 illustrates graphs displaying the mean average precision (mAP) across the number of iterations. The x-axis denotes the number of iterations, whereas the y-axis gives the value of mAP. (a) labeled mAP_0.5, signifies the mean average precision at an Intersection over Union (IoU) threshold of 0.5. IoU assesses the convergence between predicted and ground truth bounding boxes, with a 0.5 threshold implying that a predicted bounding box must overlap by at least 50% with the ground truth to be considered a correct detection. (b) labeled mAP_0.5:0.95, denotes the mean average precision averaged over IoU thresholds ranging from 0.5 to 0.95.

In essence, higher mAP values signify better prediction accuracy. Ideally, both the curves should remain high and relatively stable across all epochs. This indicates consistent model performance across various IoU thresholds throughout the training process.

Chapter 6

Conclusion

The YOLOv3 object detection model, trained and validated on a dataset comprising approximately 2000 images of floating debris collected from various sources, demonstrates significant promise in the domain of marine environment monitoring and debris detection. With a mean average precision (mAP) of 70% and an mAP at an intersection over union (IoU) threshold of 0.5 to 0.9 of 50% , the model exhibits high precision and recall rates. A precision of 0.80 indicates that the model accurately identifies debris objects, while the recall of 0.60 signifies that a significant proportion of actual debris objects are correctly identified. The model's ability to achieve a high precision rate while maintaining a substantial recall rate highlights its effectiveness in detecting floating debris in marine environments.

The loss curves for training and validation illustrate the model's learning process and improvement in predictive capabilities over epochs. The decreasing trend in box loss indicates enhanced accuracy in predicting object size and location, while the decreasing obj loss and cls loss suggest improved discernment of foreground objects from the background and accurate classification of detected debris as trash. Through the precision and recall metrics, it is evident that the model strikes a balance between identifying debris objects accurately and minimizing false positives.

In conclusion, the YOLOv3-based deep learning model presents a promising solution for real-time floating debris detection in marine environments.

6.1 Future scope

While the current model has shown promising results, there is still room for improvement. Future endeavors could focus on:

- Increasing classification accuracy: Enhancing the model to not only detect trash but also to classify each type of trash accurately.
- Improve the accuracy:By including a wide variety of dataset from different environment the model's accuracy and ability to classify different types of trash can be increased.
- Real-time Implementation: Developing a real-time implementation of the model for on-site classification of trash, enabling immediate action.

By addressing these areas, the model can be further optimized to deliver even better results, making it an indispensable tool for various applications.

References

[1] N. Shirakura, T. Kiyokawa, H. Kumamoto, J. Takamatsu and T. Ogasawara, ”Collection of Marine Debris by Jointly Using UAV-UUV With GUI for Simple Operation,” in IEEE Access, vol. 9, pp. 67432-67443, 2021

[2] M. Kremezi and V. Karathanassi, ”Data Fusion for Increasing Monitoring Capabilities of Sentinel Optical Data in Marine Environment,” in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 4809-4815, 2020

[3] Alex Solé Gómez, Leonardo Scandolo, Elmar Eisemann, ”A learning approach for river debris detection”, International Journal of Applied Earth Observation and Geoinformation, Volume 107, 2022.

[4] Li, S., Liu, S., Cai, Z. et al. ”TC-YOLOV5: rapid detection of floating debris on raspberry Pi 4B”, J Real Time Image Proc 20, volume 17, 2023.

[5] Y. Cheng et al., ”FloW: A Dataset and Benchmark for Floating Waste Detection in Inland Waters”, IEEE/CVF International Conference on Computer Vision (ICCV), pp. 10933-10942, 2021

[6] Kylili, K., Kyriakides, I., Artusi, A. et al., ”Identifying floating plastic marine debris using a deep learning approach”, Environ Sci Pollut Res 26, 17091-17099, 2019

[7] C. Lambert, M. Authier, G. Dorémus, S. Laran, S. Panigada, J. Spitz, O. Van Canneyt, V. Ridoux, ”Setting the scene for Mediterranean litterscape management: The first basin-scale quantification and mapping of floating marine debris”, Environmental Pollution, Volume 263, Part A, 2020

[8] A. Mahanti, A. Banerjee and K. Srinivasan, ”Enabling Detection of Plastic Debris on Water Surface Using Wireless Signals,” in IEEE Sensors Letters, vol. 5, no. 7, pp. 1-4, 2021

[9] Marinella Farré, ”Remote and in situ devices for the assessment of marine contaminants of emerging concern and plastic debris detection”, Current Opinion in Environmental Science & Health, Volume 18, 2020

[10] M. M. Duarte and L. Azevedo, ”Automatic Detection and Identification of Floating Marine Debris Using Multispectral Satellite Imagery,” in IEEE Transactions on Geoscience and Remote Sensing,vol. 61, pp. 1-15, 2023

[11] Srikanta Sannigrahi, Bidroha Basu, Arunima Sarkar Basu, Francesco Pilla, ”Development of automated marine floating plastic detection system using Sentinel-2 imagery and machine learning models”, Marine Pollution Bulletin, Volume 178, 2022

[12] Abdul Muhsin M, Farah F. Alkhalid, Bashra Kadhim Oleiwi, “Online Blind Assistive System using Object Recognition”, International Research Journal of Innovations in Engineering and Technology (IRJIET) ,Volume 3, Issue 12, pp 47-51, December – 2019.

[13] Aishwarya Sarkale, Kaiwant Shah, Anandji Chaudhary, Tatwadarshi P. N., “A Literature Survey: Neural Networks for Object Detection”, VIVA-Tech International Journal for Research and Innovation Volume 1, Issue 1 (2018) ISSN(Online): 2581-7280 Article No. 9.

[14] Bhumika Gupta, Ashish Chaube, Ashish Negi, Umang Goel, “Study on Object Detection using Open CV Python”, International Journal of Computer Applications Foundation of Computer Science (FCS), NY, USA, Volume 162, Number 8, 2017.

[15] Geethapriya. S, N. Duraimurugan, S.P. Chokkalingam, “Real-Time Object Detection with Yolo”, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-8, Issue-3S, February 2019.

[16] Karanbir Chahal1 and Kuntal Dey, “A Survey of Modern Object Detection Literature using Deep Learning”, International Research Journal of Engineering and Technology (IRJET), Volume 8, Issue 9, 2018.

[17] Kartik Umesh Sharma and Nileshsingh V. Thakur, “A Review and an Approach for Object Detection in Images”, International Journal of Computational Vision and Robotics, Volume 7, Number 1/2, 2017.

[18] Mukesh Tiwari, Dr. Rakesh Singhai, “A Review of Detection and Tracking of Object from Image and Video Sequences”, International Journal of Computational Intelligence Research, Volume 13, Number 5 (2017).

[19] R. Sujeetha, Vaibhav Mishra , “Object Detection and Tracking using Tensor Flow”, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8, Issue-1, May 2019.

[20] Yordanka Karayaneva and Diana Hintea, “Object Recognition in Python and MNIST Dataset Modification and Recognition with Five Machine Learning Classifiers”, Journal of Image and Graphics, Volume 6, Number 1, June 2018.

Appendix A

Program Code

```
import numpy as np
import time
import cv2
import sys
import os

CONFIG_FILE = 'custom-config.cfg'
WEIGHTS_FILE = 'best.weights'



CONFIDENCE_THRESHOLD = 0.05

LABELS = ["TRASH","TRASH"]

np.random.seed(4)
COLORS = np.random.randint(0, 255, size=(len(LABELS), 3), dtype="uint8")

net = cv2.dnn.readNetFromDarknet(CONFIG_FILE, WEIGHTS_FILE)

img_array = []
size = ()

for i in os.listdir("./tes"):
    image = cv2.imread("./tes/"+i)
    ret=True
    if ret == True:
        (H, W) = image.shape[:2]
        size = (W, H)
        ln = net.getLayerNames()
        ln = [ln[i - 1] for i in net.getUnconnectedOutLayers()]

        blob = cv2.dnn.blobFromImage(image, 1 / 255.0, (416, 416),
```

```
                                              swapRB=True,crop=False)
        net.setInput(blob)
        start = time.time()
        layerOutputs = net.forward(ln)
        end = time.time()
        print("[INFO] YOLO took {:.6f} seconds".format(end - start))

        boxes = []
        confidences = []
        classIDs = []

        for output in layerOutputs:
            for detection in output:
                scores = detection[5:]
                classID = np.argmax(scores)
                confidence = scores[classID]

                if confidence > CONFIDENCE_THRESHOLD:
                    box = detection[0:4] * np.array([W, H, W, H])
                    (centerX, centerY, width, height) =box.astype("int")

                    x = int(centerX - (width / 2))
                    y = int(centerY - (height / 2))
                    boxes.append([x, y, int(width), int(height)])
                    confidences.append(float(confidence))
                    classIDs.append(classID)

        idxs= cv2.dnn.NMSBoxes(boxes, confidences, CONFIDENCE_THRESHOLD,
                                                   CONFIDENCE_THRESHOLD)

        if len(idxs) > 0:
            for i in idxs.flatten():
                (x, y) = (boxes[i][0], boxes[i][1])
                (w, h) = (boxes[i][2], boxes[i][3])
                color = [int(c) for c in COLORS[classIDs[i]]]
                cv2.rectangle(image, (x, y), (x + w, y + h), color, 2)

                text = "{}: {:.4f}".format((LABELS[classIDs[i]]),
                                                        confidences[i])
                cv2.putText(image, text, (x, y - 5),
                                cv2.FONT_HERSHEY_SIMPLEX,0.5, color, 2)
        cv2.imshow('frame', image)
        time.sleep(2)
        #cv2.waitKey(0)
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    else:
        break
cv2.destroyAllWindows()
```

PAPER NAME

YOLOv3 report (2).pdf

AUTHOR

Pooja

WORD COUNT

6413 Words

CHARACTER COUNT

38088 Characters

PAGE COUNT

32 Pages

FILE SIZE

540.2KB

SUBMISSION DATE

May 6, 2024 1:06 PM GMT+5:30

REPORT DATE

May 6, 2024 1:07 PM GMT+5:30

11% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 7% Internet database
- 6% Publications database
- Crossref database
- Crossref Posted Content database
- 1% Submitted Works database